Timing the Message: Language-Based Notifications for Time-Critical Assistive Settings

Ya-Chuan Hsu
University of Southern California
Los Angeles, CA, USA
yachuanh@usc.edu

Jonathan DeCastro
Toyota Research Institute
Cambridge, MA, USA
jonathan.decastro@tri.global

Andrew Silva
Toyota Research Institute
Cambridge, MA, USA
andrew.silva@tri.global

Guy Rosman
Toyota Research Institute
Cambridge, MA, USA
guy.rosman@tri.global

ABSTRACT

In time-critical settings such as assistive driving, assistants often rely on alerts or haptic signals to prompt rapid human attention, but these cues usually leave humans to interpret situations and decide responses independently, introducing potential delays or ambiguity in meaning. Language-based assistive systems can instead provide instructions backed by context, offering more informative guidance. However, current approaches (e.g., social assistive robots) largely prioritize content generation while overlooking critical timing factors such as verbal conveyance duration, human comprehension delays, and subsequent follow-through duration. These timing considerations are crucial in time-critical settings, where even minor delays can substantially affect outcomes. We aim to study this inherent trade-off between timeliness and informativeness by framing the challenge as a sequential decision-making problem using an augmented-state Markov Decision Process. We design a framework combining reinforcement learning and a generated offline taxonomy dataset, balancing this trade-off while enabling a scalable taxonomy dataset generation pipeline. Empirical evaluation with synthetic humans shows our framework improves success rates by over 40% compared to methods that ignore time delays, while effectively balancing timeliness and informativeness. It also highlights an often-overlooked trade-off between these factors, opening new directions for optimizing communication in time-critical human-AI assistance.

KEYWORDS

Human-Agent Interaction, Time-Critical Settings, Language-Based Notifications

ACM Reference Format:
Ya-Chuan Hsu, Jonathan DeCastro, Andrew Silva, and Guy Rosman. 2026. Timing the Message: Language-Based Notifications for Time-Critical Assistive Settings. In *Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), Paphos, Cyprus, May 25 – 29, 2026*, IFAAMAS, 17 pages. https://doi.org/10.65109/SAWU6614

Proc. of the 25th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2026), C. Amato, L. Dennis, V. Mascardi, J. Thangarajah (eds.), May 25 – 29, 2026, Paphos, Cyprus. © 2026 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). https://doi.org/10.65109/SAWU6614

1 INTRODUCTION

Language is commonly used for coordination in embodied assistive tasks (e.g., human-human, human-robot interactions), facilitating intent communication, establishing mutual knowledge, and providing actionable instructions [23, 37, 46]. In this work, we focus specifically on time-critical scenarios, which are prevalent in daily life yet remain underexplored in language-based assistive systems. In particular, we focus on the inherent trade-off between language-based notification timeliness and informativeness. Consider a common scenario: during a highway merge, a driving assistant saying "Ease off the gas, there's a car entering from your right." provides necessary clarity but may take too long to convey and comprehend to react for collision avoidance, whereas "Slow down!" prompts rapid but potentially hazardous overreactions and results in rear-ending situations due to lack of context. Thus, effective language-based notifications in such scenarios must balance immediacy with sufficient informativeness to guide immediate and subsequent actions. This paper addresses the critical trade-off between timeliness and informativeness in the context of designing a *language-based assistive notifier agent* for *time-critical human-AI collaboration.*[1]

Prior works often emphasize providing notifications that are either timely (e.g., safety-critical alerts) [30, 47, 60] or informative (e.g., teacher-student interactions) [41, 43] independently, without explicitly addressing their trade-off. While recent works have begun recognizing that informative notifications incur time costs [31, 59], the trade-off has yet to be formalized in a sequential decision-making framework [38].

Motivated by this gap, we investigate the problem setting where the human executes a task based on their existing knowledge, the assistive agent (i.e., the *notifier*) solves a sequential decision-making task through (1) detecting knowledge gaps through human behavior, (2) determining the missing information, and (3) delivering the notification at a length that the human can comprehend before failure occurs. Additionally, while focusing on the time-critical settings, we select notifications that provide actionable instructions and that include context when given sufficient time.

To define the action space of our framework, we model each language notification as an action parameterized by three domain-agnostic properties: topic, the word-by-word point of incremental

[1]See the appendix at https://arxiv.org/pdf/2509.07438 and the code at https://github.com/SophieHsu/Timing-the-Message.

comprehension [13], and total length. To gather a dataset of notifications at scale would be impractical. For instance, a 10-word utterance would require hundreds of annotations.

Recent advances show that Large Language Models (LLMs) approximate human-like comprehension patterns [32] and replay realistic, time-ordered behavior in interactive simulations [39, 58], capturing both semantic intent and temporal dynamics. We therefore leverage LLMs as scalable surrogate annotators to infer word-level comprehension properties. This enables automatic construction of the utterance–property pairs that form our action space, overcoming the costly word-level annotation barrier to studying language-based notifications in time-critical settings.

The core contributions of this work are as follows:

- **Formal characterization of language-based notifications in time-critical human–AI collaboration.** We propose a mathematical framework for language-based assistive notifications, formalizing timeliness and informativeness for time-critical settings.
- **Reaction-time–aware decision scheme for time-critical tasks**. We provide a decision-making scheme that explicitly and robustly accounts for critical time delays, while optimizing notification timing and content to maximize safety and task efficiency.
- **Evaluation on various time-critical domains**. We demonstrate the efficacy and robustness of our approach in three domains: piloting, driving, and collaborative cooking.

2 BACKGROUND AND RELATED WORK

2.1 Language-based communications for human-robot collaboration

Collaborationfl uency in language-based human-robot collaboration (HRC) is achieved through balancing effective task-related communication and conversationalfl ow. Some prior works explored effective communication strategies such as perspective-taking for personalized delivery [55], adapting to human preferences and task states [7], providing timely reminders [14], and correcting false assumptions [43, 44]. Other works address timing implicitly through conversationalfi llers to smooth transitions without delay [41]. However, these approaches focus on relevance andfl uency [41] rather than explicitly considering the timing of information delivery to support human performance in time-critical tasks.

In time-critical collaboration, the amount of time devoted to communication must be carefully monitored and adapted. While some research acknowledges conveyance duration of assistive household agents [59], or emphasizes time-related aspects during communication, such as interruption intervals [53], they still consider communication as an instantaneous or one-timestep [52] action during planning. This simplification neglects a key trade-off: time spent on communication may delay time-critical responses. As a result, there remains a gap in explicitly modeling how communication timing interacts with task execution, especially in domains where even slight delays can lead to failure.

2.2 Human-robot collaboration in time-critical settings

We define time-critical HRC as scenarios where the effectiveness of the robot actions depends on the timing of their influence upon the human. Prior work has approached this challenge from several complementary angles. Some studies predict human intent from implicit cues such as gaze, posture, and motion [4, 22], while others model cognitive delays using drift-diffusion frameworks to capture reaction times in forced-choice tasks [18, 36], or explore aspects such as engagement [47]. At the task level, researchers analyze how robot interventions shape performance and completion time [11, 12]. In parallel, communication-focused approaches seek to optimize the timing of motion cues [28], non-verbal signals, and natural language instructions [3] to improve coordination. Although timing improves efficiency, these studies typically trade off task-execution effort against communication quantity (e.g., distance cost of motion cues, number of utterances) [53].

Beyond these trade-offs, other lines of work examine human reaction times in time-critical settings, which focus on modeling reactions to instantaneous cues within well-defined events, often in contexts such as intelligent transportation systems [36] or warning systems [25, 30, 47, 54], or designed experiments [40, 50]. In contrast, we study language-based communication in time-critical HRC, where notification delivery time can be a critical constraint. Unlike instantaneous warnings, language can convey richer guidance for human decision-making, revealing an underexplored trade-off between timeliness and informativeness.

2.3 LLMs for modeling human reasoning and behavior

Modeling human reactions is critical for human–AI collaboration. Large Language Models (LLMs) not only generate language, but also capture aspects of reasoning and behavior, demonstrating abstraction [57], common-sense reasoning [6], decision-making across domains [27, 34], and simulating feedback in interactive settings [1, 39]. They have further been used as surrogate annotators, often matching or surpassing crowdworkers in structured tasks [19, 45], and have been shown to capturefi ne-grained predictability and informativeness in comprehension [32]. Together, these results position LLMs as scalable proxies for human reasoning and annotation, motivating use as surrogate human reaction models in our framework.

3 PROBLEM DEFINITION

We consider an assistive notification task within a time-critical human-AI collaboration scenario, where the effectiveness of the assistance depends on the timeliness and informativeness of communicated notifications. In the scenario, we assume two distinct classes of heterogeneous agents with different objectives [33]: a human agent that directly affects a physical environment with their actions, and an assistive agent that delivers notifications as verbal utterances to guide or alert the human. The assistive agent's objective is to deliver notifications that not only contain actionable instructions (i.e., content that is sufficiently informative for the human to understand and act upon) but also, when appropriate, those that include additional context on how to carry out the instruction.

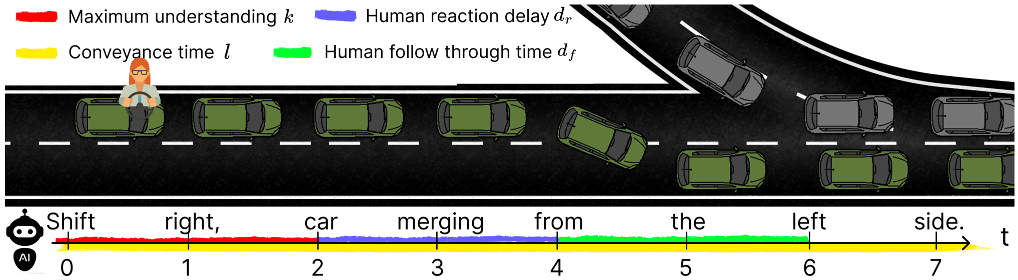

Figure 1: Illustration of a time-critical notification. An AI assistive agent notifies "Shift right, car merging from the left side" over l (yellow) time steps. The utterance has communicated actionable information at $k = 2$ (red). The human driver then has a reaction delay, d_r (blue), before beginning to move right, lasting the duration of the follow-through time, d_f (green).

For instance, to enhance the human's situational understanding or drive them towards a long-term goal.

To effectively provide decision assistance, the assistive agent needs a model of (i) how the human would solve the task unaided and (ii) how the human interprets and responds to notifications. Accordingly, the next subsection introduces a reactive human model that captures baseline task behavior and notification-driven adjustments; we then use this model to train the notifier policy.

3.1 Reactive human model

Our reactive human model, $H_{\text{react}} = (\text{MDP}, \mathcal{M}_{\text{react}})$, comprises a task-completion decision process formulated as a Markov Decision Process (MDP), together with an utterance reaction model $\mathcal{M}_{\text{react}}$.

3.1.1 Task-completion MDP. The MDP consists of a state space, S^h, action space, A^h, a stochastic transition function, T^h, describing the model's action-based state transition dynamics, a reward function, R^h, and a discount factor γ. The policy for the task-completion MDP is denoted as $\pi_{MDP}(s^h)$, where $s^h \in S^h$.

3.1.2 Utterance reaction model. We define our utterance reaction model as $\mathcal{M}_{\text{react}} = (U^g, A^h, I, R^h_{react}, \rho^{d_f}_{d_r})$, where U^g is a set of guiding utterances, where each utterance $\mathbf{u}$ is a sequence of n words, i.e., $(u_{t-n}, u_{t-n+1}, ..., u_t)$, generated by one or more assistive agent notification action a^g. The agent's behavior can thus be represented as $(a^g_{t-n}, \emptyset, ..., a^g_t)$, where most steps correspond to a null action, and occasional non-null actions produce an utterance spanning multiple words. $I : U^g \longrightarrow \mathbb{R}_{\geq 0}$ is a scalar, task-relevant informativeness measure [51], that states how much of the notification's actionable content is conveyed to the human. $I(\mathbf{u}) = 0$ indicates no action-independent details (e.g., How/Why that do not change the instructed action), and $I(\mathbf{u}') \geq I(\mathbf{u})$ whenever $\mathbf{u}'$ has more task-relevant details. R^h_{react} is the reaction reward function, and $\rho^{d_f}_{d_r}$ is a reaction function, which updates the intended human action based on the utterances and human reaction delay, d_r, and follow-through duration, d_f.

3.1.3 Task-relevant informativeness. We use a length-based proxy that measures the length of the utterance. Let l denote the length of $\mathbf{u}$ and define $I(\mathbf{u}) = l$. While the effect of informativeness may manifest in different response behavior for different tasks (see Sec. 5.2.3 for details), we model both follow-through duration and reward as a domain-specific function of informativeness, $d_f(I(\mathbf{u}))$ and $R^h_{react}(I(\mathbf{u}))$, respectively. For example, in the driving domain, d_f is a monotonic increasing function of l; for the piloting domain, we performed both constant (Appendix H.0.1) and linear (Appendix G) mappings. A brief message like "Slow down!" can trigger an immediate change in behavior, but it underspecifies the desired policy (e.g., magnitude and duration). In contrast, an instruction like "Slow down for the next few seconds" is more informative, constraining the intended action, yielding a more precisely targeted response, at the cost of some latency. Refer to Sec. 5.3.3 and Appendix H.0.1 for details on $R^h_{react}(I(\mathbf{u}))$. (See Appendix A for discussion on different types of informative notifications.)

3.1.4 Types of notifications. Since we focus on notifications that are actionable and informative, we associate a^g with a property tuple (c, k, l), where c represents the content topic, k is the *comprehension time* [15, 16], i.e., the time relative to the beginning of the utterance at which the human initiates their reaction as actionable information is conveyed. We posit $k \leq l$, where l denotes the *conveyance time*, i.e., the time that the notification takes to complete. Note that all times in this paper are measured in words, assuming an average duration of about 0.3 seconds per word [56].

3.1.5 Timing latencies. In addition to comprehension time, our human reaction model includes two temporal latencies: the reaction delay d_r, the interval between comprehension (at time k) and initiation of the instructed action, parameterized using empirical reaction-time estimates from prior work [17], and the follow-through duration d_f, the period for which the action is sustained once understood [21] (see Fig. 1).

3.1.6 Reaction function. The reaction function $\rho^{d_f}_{d_r}$ outputs the human reaction behavior after the notification is comprehended at time step $t + k$. Here, t is the absolute start of the notification action $a^g = (c, k, l)$, k is the comprehension delay relative to t, followed by a reaction delay d_r, and sustained for a duration of d_f (see Fig. 1 for an example). Let Δt denote the elapsed time since the notification was comprehended, $t' - (t + k)$, where t' is the absolute current time step. The human reaction function is defined as follows:

$$\rho^{d_f}_{d_r}(\pi_{MDP}(s_{t'}), c, \Delta t) = \begin{cases} f(c) & d_r \leq \Delta t \leq d_r + d_f \\ \pi_{MDP}(s_{t'}) & otherwise, \end{cases} \quad (1)$$

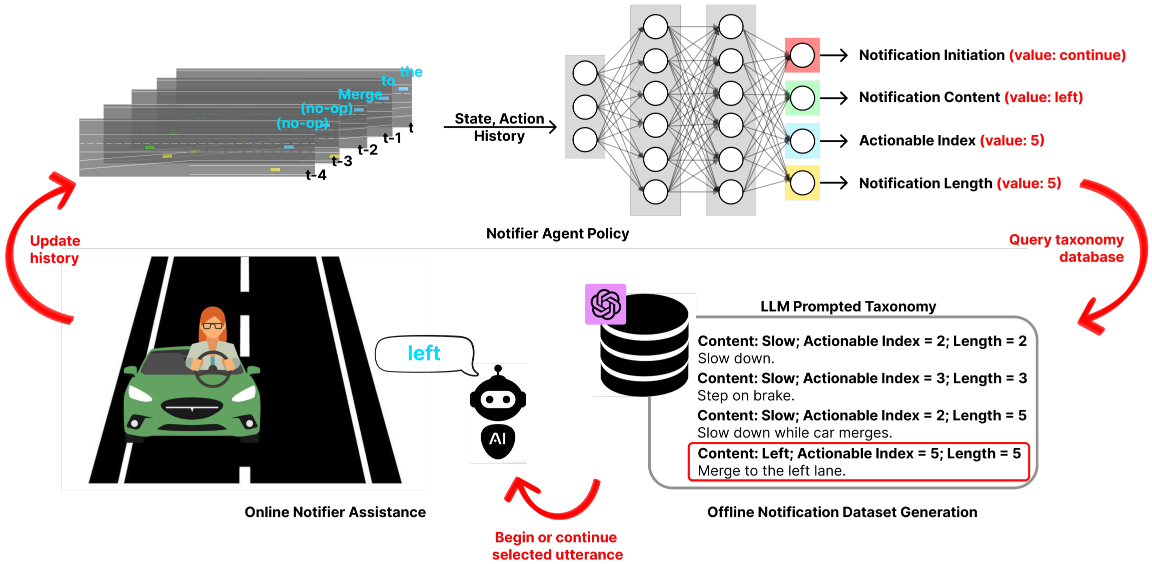

Figure 2: We present a notifier assistant that learns to provide timely information to help a human solve a task. The notifier receives recent state-action history to predict whether to provide a notification and properties of the desired notification (content, desired comprehension point, and length), which is then used to retrieve a precomputed utterance from the offline-generated utterance database matching these criteria and delivers it to the human.

where $s_{t'} \in S^h$, and $f(c)$ maps topic to the corresponding reaction a^h (e.g., sentence topic of slow down maps to deceleration actions.)

3.2 Assistive notifier model

We consider an assistive notification problem in a time-critical human-AI collaboration scenario, formulated as a sequential decision-making task. The environment consists of a model of the physical world and of the human. Because physics and human states evolve *while* an utterance is produced, we cast the assistive notification problem as an augmented-state Markov Decision Process (MDP) [48]. Following the convention from prior literature [9, 10], we define the augmented state space as afi nite history of windowed state-action pairs. Hence, we turn the delayed-feedback task into a standard MDP, a tuple $(\mathcal{X}, \mathcal{A}, T, R, \gamma)$, with the augmented state space $\mathcal{X} = (S \times \mathcal{A})^n$, $S = S^w \times S^h \times S^g$, where S^w, S^h, and S^g are the environment, human, and assistive notifier agent state spaces respectively, $\mathcal{A}$ is the action space of the assistive notifier, and n is the history window length.

At each timestep, the notifier takes an action $a_t^g \in \mathcal{A}$, which is either null (no notification) or a notification characterized by (c, k, l). If a notification is issued, the reaction function $\rho_{d_r}^{d_f}$, defined in Sec. 3.1, determines a^h by combining the utterance properties with temporal dynamics (i.e., d_r, d_f, and Δt). The state transition then decomposes as

$$T(x_{t+1}|x_t, a_t^g) = \sum_{a^h \in A^h} P(a^h|x_t, a_t^g) P_{\text{env}}(x_{t+1}|x_t, a^h),$$

where $P(a^h|x_t, a_t^g)$ is induced by $\rho_{d_r}^{d_f}$, and P_{env} captures the environment given the realized human action. The immediate reward $R(x_{t+1}, a_t^g)$ balances the task improvement and communication costs (e.g., penalties for uttering notifications). The objective is to learn an optimal notification policy $\pi^* : \mathcal{X} \rightarrow \mathcal{A}$ that maximizes the expected discounted cumulative reward.

4 LANGUAGE-BASED REACTION-AWARE NOTIFIER

To provide timely and informative assistive notification, we propose a *Language-Based Reaction-Aware Notifier* that learns to jointly reason over both *when* to communicate and *what* to communicate. The notifier comprises two interdependent components: (1) a timing-and-intent network that determines the appropriate time to deliver a notification and selects a semantic intent or category for the message, accounting for context, human comprehension time, and reaction delays; and (2) a message realization module, which queries an offline taxonomy database to retrieve an utterance that best matches the selected semantic intent. This approach enables adaptive, context-aware notification delivery that is both behaviorally informed and capable of operating in real time.

4.1 Timing-and-intent network

Our timing-and-intent network is designed to account for both human reaction delays and the temporal dynamics of message conveyance. To address these challenges, we draw on delay-aware reinforcement learning literature [9, 10], where the agent's observation is augmented with the most recent n state–action pairs. This history stack [35] formulation provides the temporal context needed to reason about decision-to-effect delays and to preempt mid-utterance by continuing, truncating, or replacing a message as conditions evolve.

For the policy architecture, prior work shows that in rapid-decision regimes, heavier encoders add inference cost with little benefit and that simple frame-stacked inputs yield stable learning in discrete, delay-aware control [2]. Guided by this, we use a lightweight MLP over the history stack and train with Proximal Policy Optimization [42]. See Appendix F for further architecture details.

4.2 Offline taxonomy generation via an LLM-surrogate reaction model

Given the optimal notification action a_t^g from the timing-and-intent network, we aim to provide utterances that correspond to the property of a_t^g. Such a process involves modeling human reactions at the word level. To scale taxonomy construction, we use LLMs as surrogate models of human reactions to notification utterances. This allows us to assign word-by-word labels to notifications for a given domain, grounded in expected human response.

Sample Prompt: Rating Comprehension Progression

Given a pilot instruction, estimate a human's comprehension level of the intended action word-by-word by
(1) Starting at 0% comprehension
(2) Reading each word left-to-right, updating comprehension:
- Key action words (e.g., "Slow") boost comprehension
...
Output a list of comprehension values (0–100%) after each word per instruction.

Example ("Slow down" type):
Immediate | speed | reduction | needed | danger!
5% | 20% | 80% | 80% | 100%
Adjust | speed | prepare | to | avoid | the | zone.
5% | 50% | 50% | 50% | 70% | 70% | 70%

We build on LLMs' ability to approximate human comprehension [20, 45] and prompt LLMs to: (1) generate notifications for a topic c, and (2) assign comprehension progression scores to each word [61], identifying the word index k where a human might react [16]. The generated notifications are then categorized by their properties. (This is done entirely offline; at test time, we use only a compact policy network and a database lookup.) A sample prompt for (2) is shown above; see Appendix I for more details. For this paper, these numeric comprehension sequences are illustrative, and we leave calibration to actual humans as future work.

4.3 Notifier and human interaction

As detailed in Sec. 3.2, the notifier interacts with the human. At each timestep t, the network selects a notification action a_t^g that maximizes the cumulative reward. This action yields either null or a notification characterized by (c, k, l). The notifier then outputs either null or a word drawn from the notification queried from the offline-generated taxonomy database indexed by (c, k, l). See Fig. 2 for an illustration of how the notifier selects and delivers the notification. Note that, as delivery is word-by-word, the human may begin acting after the earliest actionable prefix k is understood (possibly before the full message completes) (see Sec. 5.3.4).

5 EXPERIMENTS

5.1 Domains

We evaluate our approach in time-critical domains that lend themselves to assistance, where the human lacks certain task-relevant information that the assistant can access. We modify three domains for this purpose: Lunar Lander [5], Highway Merging [29], and Steakhouse [24]. In each, the human is under time pressure, and lacks certain information, offering complementary testbeds for evaluating notification design.

5.1.1 Lunar Lander (Piloting). This domain [5] involves a human piloting a spacecraft to land on a designated platform, while navigating a dynamic environment containing danger zones, which are visible to the notifier but occluded from the pilot. Due to the rapidly evolving and control-intensive nature of the task, the notifier must provide timely notifications, helping the pilot quickly adjust to stay outside of danger zones, remain in control of the spacecraft, and land safely.

5.1.2 Highway Merging (Driving). This domain, modified from [29], features a human-driven ego vehicle on a highway with three merging events, adapted from [29]. Merging vehicles are unaware of the ego vehicle, while the human driver expects them to yield. The notifier's objective is to proactively alert the driver to maneuver away from impending collision risks, thereby encouraging smooth trafficfl ow free of unnecessary decelerations or abrupt maneuvers.

5.1.3 Steakhouse (Cooking). This domain [24], adapted from [8], simulates a community kitchen where a human collaborates with a notifier to complete subtasks (e.g., cooking meat, chopping radishes, washing dirty plates, and plating cooked steak garnished with chopped radishes). The human has partial observability—some stations are already occupied and only revealed upon approach—while the notifier has full observability. The notifier proactively issues concise prompts, "On your left", to encourage quick action interventions, or incremental actionable notifications like "Shift left, all the stations are occupied", to enable quick interventions and support the human's mental model (see Appendix E for details). This setup illustrates how timely context-aware notifications aid both immediate actions and broader decision-making.

5.2 Experimental setup

5.2.1 Notifier Architecture. The notifier uses a feedforward neural network trained on domain-specific processed observations (details in Appendix F) and outputs separate categorical distributions for (1) notification initiation, (2) topic c, (3) comprehension time k, and (4) length l, each predicted through a distinct linear projection from shared features. The initiation projection determines whether to notify, while the remaining projections define the notification action $a^g = (c, k, l)$.

5.2.2 Notification types. Apart from setting the notification action as above, we consider three types of notification actions. Each type varies in notification length and informativeness, defining different bounds of the action space $\mathcal{A}$.

- **Topic-only notifications** convey short, minimal, high-urgency commands without elaboration (e.g., "Stop!")

Table 1: Success rates for piloting and driving (primary metric) with secondary metrics, notification frequency and follow-through rate, illustrating trade-offs from considering human reaction delays. Blue indicates the best success rate.

Notifier Policy	Success Rate (↑)		Secondary Metrics			
	Piloting	Driving	Piloting Noti.	Piloting Follow	Driving Noti.	Driving Follow
Heuristic	0.00	N/A	0.50	0.76	N/A	N/A
Delay-Free Notifier	0.22 ± 0.04	0.28 ± 0.03	0.25 ± 0.01	0.83 ± 0.03	0.58 ± 0.18	0.04 ± 0.02
Notifier w/ Convey	0.94 ± 0.03	0.87 ± 0.11	0.22 ± 0.01	0.82 ± 0.01	0.22 ± 0.10	0.32 ± 0.13
Convey & React (Ours)	0.97 ± 0.02	0.93 ± 0.02	0.22 ± 0.01	0.82 ± 0.01	0.35 ± 0.19	0.18 ± 0.14

Table 2: Notifier policy robustness to human reaction delays d_r (timesteps) in the Lunar Lander domain. The policy remains robust for d_r values below the training value. Blue highlights indicate performance that either matches or outperforms the Matching condition (human d_r aligned with training d_r).

Reaction	Avg. Noti. Freq. (↓)			Avg. Follow-Through Rate (↑)			Success Rate (%) (↑)		
	$\mathcal{N}(2, 0.5)$	$\mathcal{N}(2, 1.0)$	Matching	$\mathcal{N}(2, 0.5)$	$\mathcal{N}(2, 1.0)$	Matching	$\mathcal{N}(2, 0.5)$	$\mathcal{N}(2, 1.0)$	Matching
$d_r = 0$	0.21 ± 0.00	0.21 ± 0.01	0.22 ± 0.01	0.68 ± 0.03	0.69 ± 0.06	0.82 ± 0.01	0.97 ± 0.01	0.96 ± 0.02	0.94 ± 0.03
$d_r = 1$	0.21 ± 0.01	0.21 ± 0.01	0.22 ± 0.01	0.68 ± 0.04	0.68 ± 0.06	0.71 ± 0.05	0.98 ± 0.00	0.98 ± 0.01	0.98 ± 0.01
$d_r = 2$	0.21 ± 0.01	0.21 ± 0.01	0.22 ± 0.01	0.81 ± 0.01	0.83 ± 0.01	0.82 ± 0.01	0.96 ± 0.02	0.96 ± 0.02	0.97 ± 0.02
$d_r = 3$	0.42 ± 0.02	0.42 ± 0.00	0.27 ± 0.01	0.04 ± 0.02	0.05 ± 0.05	0.88 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	0.11 ± 0.06
$d_r = 4$	0.46 ± 0.00	0.46 ± 0.00	0.15 ± 0.01	0.01 ± 0.00	0.01 ± 0.02	0.28 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	0.10 ± 0.02

- **Complete-utterance notifications** assume the human comprehends only after the entire notification is delivered ($k = l$) (e.g., "Once past the merge point, slow down.").
- **Incrementally actionable notifications** are designed for partial comprehension during delivery ($k \le l$) that is sufficient to trigger a response, with the remaining content adding clarifying context. For instance, "Slow down to avoid the left merging vehicle" prompts action upon hearing "Slow down," with the rest of the notification providing context.

5.2.3 Notification informativeness. Across all domains, we assume the informativeness of a notification $\mathbf{u}$ is a monotone non-decreasing function of its message length l, i.e., $I(\mathbf{u}) = f(l)$ with f monotone. Its effects manifest differently depending on the task: in piloting, informativeness improves task understanding and stability, which we capture by defining the human reward as $R^h_{react}(I(\mathbf{u}))$; in driving, it modulates reaction behavior through follow-through duration $d_f(I(\mathbf{u}))$, where more informative notifications sustain actions for longer; and in cooking, it conveys station occupancy information, thereby updating the human's environment state and shaping subsequent decision-making.

5.2.4 Human agent. The task-completion MDP policy is trained separately for each domain to capture environment-specific dynamics (Appendix D). In contrast, all notifiers are evaluated under identical parameter settings for the reaction function $\rho^{d_f}_{d_r}$, ensuring consistent notification–reaction dynamics. The only exception is the robustness analysis study (Section 5.3.2), where reaction parameters are varied to examine the effects of model mismatch.

5.2.5 Evaluation metrics. Our primary metric is success rate: the fraction of episodes that achieve the task goal while satisfying all safety constraints (reported as a percentage). To characterize notification behavior we report three secondary metrics. First, the *notification frequency*: let E be the number of episodes, T_e the number of environment time steps in episode e, and N_e the number of *delivered* (non-null) notifications in episode e. We report $\frac{1}{E}\sum_{e=1}^{E}\frac{N_e}{T_e}$.

Second, the *follow-through rate*: for each delivered notification $d \in \mathcal{D}$ shown at time step t_d, we define: (i) a *comprehension* indicator $c_d = 1$ if the human comprehends the notification; otherwise $c_d = 0$; (ii) an *action* indicator $a_d = 1$ if, within the reaction time window after t_d, the user executes the notified action and was not already in progress at t_d; otherwise $a_d = 0$. The follow-through rate is $\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}} \mathbf{1}[\, c_d = 1 \wedge a_d = 1\,]$.

Lastly, the *long notification rate*. Let $\ell(d)$ be the word count of the notification d. A notification is long if $\ell(d) \ge L$; we use L=5 words. The rate is $\frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}} \mathbf{1}[\, \ell(d) \ge L\,]$. All metrics are averaged over episodes and random seeds.

5.2.6 Baselines. We compare our method against three baselines that differ in how they account for notification timing, while holding the human reaction modelfi xed across domains. This isolates the impact of explicitly modeling conveyance and reaction delays in the notifier's decision-making.

- **Heuristic Notifier**: Issues alerts when the human approaches a danger zone, directing them to follow a predefined safe path once a distance threshold is crossed.
- **Delay-Free Notifier (RL) [31]**: Assumes notifications are delivered and acted upon instantaneously ($k = 0$, $d_r = 0$), ignoring both conveyance and reaction delays.
- **Notifier With Conveyance Time Awareness (RL) [59]**: Models the time required to deliver a notification ($k > 0$), but assumes immediate human response ($d_r = 0$).

5.2.7 Our method. **Notifier With Conveyance and Reaction Time Awareness**: Incorporates both notification conveyance duration and human reaction delays, explicitly modeling a cumulative two-timestep delay ($d_r = 2$) [17].

5.3 Results

We organize our experiments to answer the following:

(Q1) Does a conveyance and reaction time aware notifier result in significant performance improvements over baseline approaches?

(Q2) Can a notifier trained on a limited distribution of reaction times generalize to out-of-distribution cases?

(Q3) Does conveyance and reaction time aware notifier balance notification timeliness and informativeness to improve overall task performance?

5.3.1 Performance analysis (Q1). We evaluate notifier effectiveness primarily using success rate, which reflects both task safety and goal achievement in time-critical settings. As shown in Table 1, our proposed Convey & React notifier, explicitly modeling both conveyance and reaction delays, achieves the highest success rates in piloting (0.97) and driving (0.93). By contrast, the Delay-Free baseline performs poorly, highlighting the risks of ignoring timing altogether. The w/ Convey baseline, which accounts only for conveyance delays, already achieves strong performance, indicating that modeling conveyance alone yields substantial gains.

To assess statistical reliability, we evaluate each method over 5 seeds (100 evaluation episodes per seed) and run paired t-tests with Holm–Bonferroni correction. Convey & React significantly improves over Delay-Free in both Highway and Lunar Lander (Holm $p<0.01$), while the additional gains over Notifier w/ Convey are not significant (Holm $p=0.16/0.14$), suggesting that modeling conveyance time provides the most benefit in these domains. We expect reaction-delay modeling to matter most when messages are aggressively pruned to be short, as small differences in delivery and interpretation time can affect downstream actions (e.g., team sports). Overall, these results show that explicitly modeling conveyance and reaction delays is key for time-critical assistive performance.

5.3.2 Robustness analysis (Q2). Human responses to notifications vary due to differences in comprehension, reaction delay, and follow-through. In our formulation, these sources of variability enter through the reaction function $\rho_{d_r}^{d_f}$, which determines the lag between a notification and observed behavior. For tractability, we vary the reaction delay d_r, since robustness to unseen delays directly tests generalization to out-of-distribution timing, while variability in comprehension or follow-through would enter through the same functions ($\rho_{d_r}^{d_f}$ or $d_f(I(\mathbf{u}))$) and are left for future work.

Table 2 compares two training regimes: a *population-trained* notifier, which samples d_r from a Gaussian distribution $\mathcal{N}(2, \sigma^2)$, and a *matching* notifier trained with afi xed delay equal to evaluation (upper bound). For the population setting we sample once per episode while $d_r = \text{clip}(\text{round}(d_r), 0, 4)$, and hold d_r fixed for all notifications within that episode. At evaluation we deterministically set $d_r \in \{0, \dots, 4\}$. The population-trained notifier generalizes well to unseen delays shorter than those sampled during training,

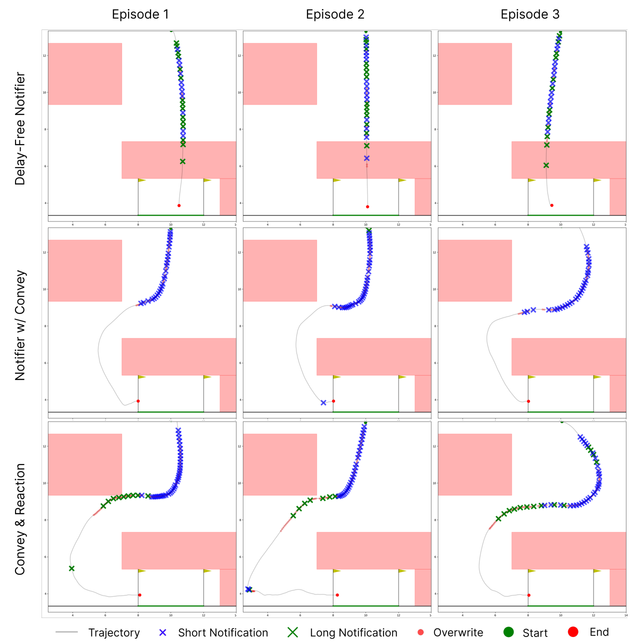

Figure 3: Trade-off between notification timeliness and informativeness across policies in the Lunar Lander (Harder Version) domain. Gray shows trajectory (start: green, end: red); shaded regions are hazards. Notifications are short (blue ×) or long (green ×); red markers (•) indicate overwritten messages. Effective policies send early short alerts to buy time, then switch to longer, more informative notifications once time-to-collision increases. Delay-Free Notifier (top) assumes instant comprehension andfi res nearly every step, so evenly spaced alerts arrive too late. Notifier w/ Convey (middle) is conservative, clustering early short alerts and keeping them while hovering over the central hazard. Convey & React (bottom) starts with short warnings, then shifts to longer notifications as descent slows and reaction time becomes available; all stop once descent is safe without assistance.

achieving success rates near the upper bound (0.96–0.98). However, both population-trained and matching policies' performance drop once delays become very long ($d_r \geq 3$), reflecting a fundamental limitation: excessively delayed human reactions cannot be fully compensated by notification policies.

5.3.3 Timeliness and informativeness analysis (Q3). We examine how notifiers balance timeliness and informativeness by comparing Topic-Only notifications—short prompts that trigger immediate actions (e.g., "Speed up!")—and Complete-Utterance notifications—longer messages that sustain actions (e.g., "Speed up for 5 seconds"). In ourfi rst experiment, informativeness is linked to follow-through duration $d_f(I(u))$ (Sec. 3.1). Table 3 shows that in the piloting domain, both notification types reach a success rate of 0.97, showing that highly dynamic environments leave little room for sustained actions. In the driving domain, however, Complete-Utterance improves success (0.97 vs. 0.93) by sustaining driver responses. Although success rate shows a benefit, secondary metrics

Table 3: Performance of different notification types in piloting and driving. Success rates are shownfi rst as the primary metric; notification frequency and follow-through rate are secondary metrics. Blue indicates the best success rate.

Notifier Policy	Success Rate (%) ↑		Avg. Noti. Freq.		Avg. Follow-Through Rate		Long Noti. Rate	
	Piloting	Driving	Piloting	Driving	Piloting	Driving	Piloting	Driving
Topic Only ($\langle c \rangle$)	0.97 ± 0.02	0.93 ± 0.04	0.22 ± 0.01	0.20 ± 0.09	0.77 ± 0.02	0.32 ± 0.09	N/A	N/A
Complete-Utterance ($\langle c, l \rangle$)	0.97 ± 0.02	0.97 ± 0.03	0.22 ± 0.01	0.37 ± 0.12	0.82 ± 0.01	0.29 ± 0.04	0.01 ± 0.01	0.37 ± 0.23

Figure 4: In Steakhouse domain, the notification "Go down, all stations occupied" (denoted [2 1 5] in thefi gures) begins at t = 0. Mid-sentence, the human comprehends the initial instruction and starts moving downward. By t = 5, upon full delivery and comprehension, the human updates their mental model and proceeds with optimal actions based on the current kitchen state.

reveal a trade-off: longer utterances are often interrupted by state changes, which forces preemption and lowers follow-through on notifications that were not fully delivered.

The saturated success rate in piloting suggests that informativeness, when defined through follow-through, has a limited effect in highly dynamic environments, with benefits more likely reflected in dimensions not captured by success rate, such as communication efficiency, collaborationfl uency, or trust. Hence, we run a second experiment where informativeness is tied to reward rather than follow-through, allowing us to observe how policies reallocate toward longer messages while holding compliancefi xed. Additionally, we study a harder version of the original lunar lander environment (see Appendix H for details) to further evaluate the trade-off. Figure 3 shows that the Delay-Free notifier issues long utterances too late, while the w/ Convey notifier improves safety with short, timely alerts, but as it assumes zero reaction delay, it avoids longer messages that risk late interventions. In contrast, our Convey & React notifier adapts by beginning with short alerts to prevent immediate failure, then switching to longer utterances once the lander stabilizes, and stopping once passed danger zones.

Together, these results demonstrate that explicitly modeling both conveyance and reaction delays enables notifiers to adapt notification length to task conditions—favoring timeliness when urgent and informativeness when time permits.

5.3.4 Incrementally actionable notifications. We also demonstrate incrementally actionable notifications in the Steakhouse cooking domain (Fig. 4), where notifications are structured with an initial action cue followed by additional context. For example, the utterance "Go down, all stations occupied" begins at $t = 0$, by $t = 2$ the human has already acted on the initial cue ("Go down"), and by $t = 5$ the full message ("all stations occupied") updates the human's situational awareness for future decisions (see Appendix E for details). This demonstration shows how incrementally actionable notifications can prevent immediate errors while providing context that supports longer-term planning. Although they offer limited benefit in fast dynamics domains such as piloting or driving, they highlight the value of informativeness in domains like cooking, where success depends on balancing short-term responses with future coordination.

6 CONCLUSION AND DISCUSSION

In this paper, we propose a framework for optimizing the timing and content of language-based notifications in AI-assisted interactions, explicitly modeling instruction duration, human comprehension delay, and subsequent response. By formulating the problem as a sequential decision-making task and applying reinforcement learning, our method outperforms baselines that assume instantaneous human reactions or account only for notification delivery time. While our evaluation relies on LLMs to simulate comprehension and draws from prior work on reaction times, validating these assumptions with human subjects and developing mechanisms to handle, e.g. variations due to fatigue, remain important directions to explore. Another promising avenue is refining the offline taxonomy dataset with human-in-the-loop feedback. Overall, our results show that accounting for both informativeness and timeliness is not only feasible but impactful, and this work is expected to serve as a foundation for more human-aligned, adaptive communication in time-critical settings.

REFERENCES

[1] Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. 2023. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning*. PMLR, 337–371.

[2] Ivan Anokhin, Rishav Rishav, Matthew Riemer, Stephen Chung, Irina Rish, and Samira Ebrahimi Kahou. 2025. Handling Delay in Real-Time Reinforcement Learning. *arXiv preprint arXiv:2503.23478* (2025).

[3] Jimmy Baraglia, Maya Cakmak, Yukie Nagai, Rajesh PN Rao, and Minoru Asada. 2017. Efficient human-robot collaboration: when should a robot take initiative? *The International Journal of Robotics Research* 36, 5-7 (2017), 563–579.

[4] Cynthia Breazeal, Cory D Kidd, Andrea Lockerd Thomaz, Guy Hoffman, and Matt Berlin. 2005. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *International conference on intelligent robots and systems*. IEEE, 708–713.

[5] Greg Brockman, Vicki Cheung, Ludwig Pettersson, et al. 2016. OpenAI Gym. *arXiv preprint arXiv:1606.01540* (2016).

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems* 33 (2020), 1877–1901.

[7] Guilhem Buisan, Guillaume Sarthou, and Rachid Alami. 2020. Human aware task planning using verbal communication feasibility and costs. In *International Conference on Social Robotics*. Springer, 554–565.

[8] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. 2019. On the utility of learning about humans for human-ai coordination. *Advances in Neural Information Processing Systems* 32 (2019).

[9] Baiming Chen, Mengdi Xu, Liang Li, and Ding Zhao. 2021. Delay-aware model-based reinforcement learning for continuous control. *Neurocomputing* 450 (2021), 119–128.

[10] Baiming Chen, Mengdi Xu, Zuxin Liu, Liang Li, and Ding Zhao. 2020. Delay-aware multi-agent reinforcement learning for cooperative and competitive environments. *arXiv preprint arXiv:2005.05441* (2020).

[11] Kaiqi Chen, Jeffrey Fong, and Harold Soh. 2022. MIRROR: Differentiable Deep Social Projection for Assistive Human-Robot Communication. In *Proceedings of Robotics: Science and Systems*.

[12] Rohan Choudhury, Gokul Swamy, Dylan Hadfield-Menell, and Anca D Dragan. 2019. On the utility of model learning in HRI. In *International Conference on Human-Robot Interaction*. IEEE, 317–325.

[13] Monika Chromik, Birte Carlmeyer, and Britta Wrede. 2017. Ready for the Next Step? Investigating the Effect of Incremental Information Presentation in an Object Fetching Task. In *Companion of the ACM/IEEE International Conference on Human-Robot Interaction*. Association for Computing Machinery, 95–96. https://doi.org/10.1145/3029798.3038352

[14] Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. 2019. TarMAC: Targeted multi-agent communication. In *International Conference on Machine Learning*. PMLR, 1538–1546.

[15] David DeVault, Kenji Sagae, and David Traum. 2009. Can Ifi nish? Learning when to respond to incremental interpretation results in interactive dialogue. In *SIGDIAL*. 11–20.

[16] David DeVault, Kenji Sagae, and David Traum. 2011. Incremental interpretation and prediction of utterance meaning for interactive dialogue. *Dialogue & Discourse* 2, 1 (2011), 143–170.

[17] Paweł Droździel, Sławomir Tarkowski, Iwona Rybicka, and Rafał Wrona. 2020. Drivers' reaction time research in the conditions in the real traffic. *Open Engineering* 10, 1 (2020), 35–47.

[18] Drew Fudenberg, Whitney Newey, Philipp Strack, and Tomasz Strzalecki. 2020. Testing the drift-diffusion model. *Proceedings of the National Academy of Sciences* 117, 52 (2020), 33141–33148.

[19] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences* 120, 30 (2023), e2305016120.

[20] Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences* 120, 30 (2023), e2305016120. https://doi.org/10.1073/pnas.2305016120 arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.2305016120

[21] Peter M Gollwitzer. 1999. Implementation intentions: strong effects of simple plans. *American psychologist* 54, 7 (1999), 493.

[22] Guy Hoffman, Tapomayukh Bhattacharjee, and Stefanos Nikolaidis. 2023. Inferring human intent and predicting human action in human–robot collaboration. *Annual Review of Control, Robotics, and Autonomous Systems* 7 (2023).

[23] Guy Hoffman and Cynthia Breazeal. 2004. Collaboration in human-robot teams. In *AIAA 1st intelligent systems technical conference*. 6434.

[24] Ya-Chuan Hsu, Michael Defranco, Rutvik Patel, and Stefanos Nikolaidis. 2025. Integrating Field of View in Human-Aware Collaborative Planning. In *International Conference on Robotics and Automation*.

[25] I Ge Jin and Gábor Orosz. 2016. Optimal control of connected vehicle systems with communication delay and driver reaction time. *IEEE Transactions on Intelligent Transportation Systems* 18, 8 (2016), 2056–2070.

[26] Arne Kesting, Martin Treiber, and Dirk Helbing. 2007. General lane-changing model MOBIL for car-following models. In *Transportation Research Board 86th Annual Meeting*.

[27] Martin Klissarov, Pierluca D'Oro, Shagun Sodhani, Roberta Raileanu, Pierre-Luc Bacon, Pascal Vincent, Amy Zhang, and Mikael Henaff. 2023. Motif: Intrinsic motivation from artificial intelligence feedback. *arXiv preprint arXiv:2310.00166* (2023).

[28] Kin Man Lee, Arjun Krishna, Zulfiqar Zaidi, Rohan Paleja, Letian Chen, Erin Hedlund-Botti, Mariah Schrum, and Matthew Gombolay. 2023. The Effect of Robot Skill Level and Communication in Rapid, Proximate Human-Robot Collaboration. In *International Conference on Human-Robot Interaction*.

[29] Edouard Leurent. 2018. An Environment for Autonomous Driving Decision-Making. https://github.com/eleurent/highway-env.

[30] Chenran Li, Aolin Xu, Enna Sachdeva, Teruhisa Misu, and Behzad Dariush. 2024. Optimal Driver Warning Generation in Dynamic Driving Environment. In *International Conference on Robotics and Automation*. IEEE, 14184–14190.

[31] Jijia Liu, Chao Yu, Jiaxuan Gao, Yuqing Xie, Qingmin Liao, Yi Wu, and Yu Wang. 2024. LLM-powered hierarchical language agent for real-time human-ai coordination. *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)* (2024).

[32] Adrielli Tina Lopes Rego, Joshua Snell, and Martijn Meeter. 2024. Language models outperform cloze predictability in a cognitive model of reading. *PLOS Computational Biology* 20, 9 (2024), e1012117.

[33] Matthew B Luebbers, Aaquib Tabrez, Kyler Ruvane, and Bradley Hayes. 2023. Autonomous Justification for Enabling Explainable Decision Support in Human-Robot Teaming.. In *Robotics: Science and Systems*.

[34] Zhao Mandi, Shreeya Jain, and Shuran Song. 2024. Roco: Dialectic multi-robot collaboration with large language models. In *International Conference on Robotics and Automation*. IEEE, 286–299.

[35] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[36] Samir HA Mohammad, Haneen Farah, and Arkady Zgonnikov. 2024. In the driver's mind: Modeling the dynamics of human overtaking decisions in interactions with oncoming automated vehicles. *Transportation Research Part F: Traffic Psychology and Behaviour* 107 (2024), 562–577.

[37] Stefanos Nikolaidis, Minae Kwon, Jodi Forlizzi, and Siddhartha Srinivasa. 2018. Planning with verbal communication for human-robot collaboration. *ACM Transactions on Human-Robot Interaction* 7, 3 (2018), 1–21.

[38] Ravi Pandya, Madison Bland, Duy P Nguyen, Changliu Liu, Jaime Fernández Fisac, and Andrea Bajcsy. 2025. From Refusal to Recovery: A Control-Theoretic Approach to Generative AI Guardrails. *arXiv preprint arXiv:2510.13727* (2025).

[39] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *ACM symposium on user interface software and technology*. 1–22.

[40] Roger Ratcliff, Cynthia Huang-Pollock, and Gail McKoon. 2018. Modeling individual differences in the go/no-go task with a diffusion model. *Decision* 5, 1 (2018), 42.

[41] Merle M Reimann, Florian A Kunneman, Catharine Oertel, and Koen V Hindriks. 2024. A survey on dialogue management in human-robot interaction. *ACM Transactions on Human-Robot Interaction* 13, 2 (2024), 1–22.

[42] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[43] Maayan Shvo, Ruthrash Hari, Ziggy O'Reilly, Sophia Abolore, Sze-Yuh Nina Wang, and Sheila A McIlraith. 2022. Proactive Robotic Assistance via Theory of Mind. In *International Conference on Intelligent Robots and Systems (IROS)*. 9148–9155.

[44] Maayan Shvo, Toryn Q Klassen, and Sheila A McIlraith. 2022. Resolving misconceptions about the plans of agents via Theory of Mind. In *Proceedings of the International Conference on Automated Planning and Scheduling*, Vol. 32. 719–729.

[45] Siddharth Srikanth, Varun Bhatt, Boshen Zhang, Werner Hager, Charles Michael Lewis, Katia P Sycara, Aaquib Tabrez, and Stefanos Nikolaidis. 2025. Algorithmic Prompt Generation for Diverse Human-like Teaming and Communication with Large Language Models. *arXiv preprint arXiv:2504.03991* (2025).

[46] Aaron St. Clair and Maja Mataric. 2015. How robot verbal feedback can improve team performance in human-robot task collaborations. In *International Conference on Human-Robot Interaction*. 213–220.

[47] Rashmi Sundareswara, Mike Daily, Mike Howard, Howard Neely, Rajan Bhattacharyya, and Craig Lee. 2013. Using a distracted driver's behavior to inform the timing of alerts in a semi-autonomous car. In *International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support*. IEEE, 199–202.

[48] Richard S Sutton, Andrew G Barto, et al. 1998. *Reinforcement learning: An introduction*. Vol. 1. MIT press Cambridge.

[49] Martin Treiber, Andreas Hennecke, and Dirk Helbing. 2000. Congested traffic states in empirical observations and microscopic simulations. *Transportation*

Science 39, 2 (2000), 147–159.
[50] Jennifer S Trueblood, Michael J Endres, Jerome R Busemeyer, and Peter R Finn. 2011. Modeling response times in the go/no-go discrimination task. In *Cogsci*, Vol. 2011. 1866.
[51] Mycal Tucker, Julie Shah, Roger Levy, and Noga Zaslavsky. 2025. Towards Human-Like Emergent Communication via Utility, Informativeness, and Complexity. *Open Mind* 9 (04 2025), 418–451. https://doi.org/10.1162/opmi_a_00188 arXiv:https://direct.mit.edu/opmi/article-pdf/doi/10.1162/opmi$_{a0}$0188/2512659$/opmi_{a0}0188.pdf$
[52] Vaibhav Unhelkar and Julie Shah. 2016. Contact: Deciding to communicate during time-critical collaborative tasks in unknown, deterministic domains. In *AAAI Conference on Artificial Intelligence*, Vol. 30.
[53] Vaibhav V Unhelkar, Shen Li, and Julie A Shah. 2020. Decision-making for bidirectional communication in sequential human-robot collaborative tasks. In *International Conference on Human-Robot Interaction*. 329–341.
[54] Jie Wang, Yash Vardhan Pant, Lei Zhao, Michał Antkiewicz, and Krzysztof Czarnecki. 2024. Enhancing safety in mixed traffic: Learning-based modeling and efficient control of autonomous and human-driven vehicles. *IEEE Transactions on Intelligent Transportation Systems* (2024).
[55] Yuanfei Wang, Fangwei Zhong, Jing Xu, and Yizhou Wang. 2022. Tom2c: Target-oriented multi-agent communication and cooperation with theory of mind. *ICLR* (2022).
[56] Jiahong Yuan, Mark Liberman, and Christopher Cieri. 2006. Towards an integrated understanding of speaking rate in conversation. In *International Conference on Spoken Language Processing*.
[57] Ceyao Zhang, Kaijie Yang, Siyi Hu, Zihao Wang, Guanghe Li, Yihang Sun, Cheng Zhang, Zhaowei Zhang, Anji Liu, Song-Chun Zhu, et al. 2024. Proagent: building proactive cooperative agents with large language models. In *AAAI Conference on Artificial Intelligence*, Vol. 38. 17591–17599.
[58] Erhan Zhang, Xingzhu Wang, Peiyuan Gong, Yankai Lin, and Jiaxin Mao. 2024. Usimagent: Large language models for simulating search users. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2687–2692.
[59] Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B Tenenbaum, Tianmin Shu, and Chuang Gan. 2024. Building cooperative embodied agents modularly with large language models. *ICLR* (2024).
[60] Xuan Zhao, Tingxiang Fan, Yanwen Li, Yu Zheng, and Jia Pan. 2021. An efficient and responsive robot motion controller for safe human-robot collaboration. *IEEE Robotics and Automation Letters* 6, 3 (2021), 6068–6075.
[61] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems* 36 (2023), 46595–46623.